The Planets IF - A Framework for Integrated Access to Preservation Tools

Rainer Schmidt, Andrew Lindley, Ross King
Austrian Institute of Technology
Vienna, Austria
first.lastname@ait.ac.at

Andrew Jackson, Carl Wilson
The British Library
Boston Spa, West Yorkshire, UK
first.lastname@bl.uk

Fabian Steeg
University of Cologne
Albertus-Magnus-Platz, Cologne, Germany
first.lastname@uni-koeln.de

ABSTRACT

The Planets project is driven by requirements for the long-term preservation faced by institutional libraries and archives. The project develops an integrated environment that allows archivists to seamlessly utilize and evaluate tools and strategies for the preservation of cultural heritage data. The Planets Interoperability Framework (IF) supports this vision by providing the technical backbone for integrating existing content repositories, preservation tools, and services into a service-oriented research infrastructure. It implements a number of common software components for user authentication, data access, or service orchestration. Moreover, it defines the interfaces and communication protocols for preservation services like identification, characterization, migration or rendering. It thereby assures the interoperability of the various heterogeneous preservation tools and applications in order to establish a coherent and extensible preservation system. In this paper, we present the service architecture as well as the runtime environment and its application.

1. INTRODUCTION

There is a vital need to ensure long-term access to the growing collections of digital data across almost all areas of society [14]. In addition to the physical preservation of the content bit-streams, one must ensure the interpretability of the digital objects with current and future applications in order to prevent a loss of information. Consequently, digital preservation imposes a major challenge for the development of digital library and archive information systems [7]. The development of preservation strategies and automated workflows provides a major research goal in this area. The EU project Planets aims to provide a service-based research environment that addresses the digital preservation challenges that digital libraries and archives are facing [5]. The system provides a web portal that integrates existing repositories of cultural heritage institutions and a large number of preservation tools allowing data curators to conduct and evaluate preservation experiments.

The Interoperability Framework (IF) provides the technical framework that governs the integration of the end user applications with preservation services and data repositories. It implements a number of common web services like *authentication*, *data access*, *workflow enactment*, and defines the service interfaces for the preservation components like *migration* and *characterization*. This architecture conforms to the concept of a Science Gateway [9] allowing users to easily access and utilize distributed resources through a common Web Portal interface. Such gateways support access to orchestrated workflows, computational resources, information services and data directories [19]. An important benefit of this approach originates from the functionality as a gateway to different backend systems. A flexible service-oriented gateway architecture that is designed to be run on desktop computers as well as interoperate with a solid grid infrastructure is described by Gannon et al. [6].

A central component is provided by the Planets IF workflow engine providing a Web service API for the submission, execution, and monitoring of preservation workflows. Workflow documents can be assembled from a set of high-level components that implement a specific preservation functionality and abstract away the underlying messaging layer. Components that perform preservation actions often rely on resource-intensive operations and pre-installed tools (e.g. a file format converter) that are wrapped and remotely executed via web services. A preservation experiment typically includes the execution of a tool chain in order to perform tasks like metadata extraction and mapping, data migration and comparison, or knowledge and information extraction. Key aspects of the system are the integration of distributed repositories, the development of the required preservation services and interfaces, a consistent technical vocabulary and registry foundation, and workflows that capture and evaluate results and preservation metadata.

The paper is organized as follows: Section 2 provides an introduction to the problem and presents an example use case, section 3 describes the system architecture, section 4 outlines the workflow environment, section 5 surveys related work, and section 6 concludes the paper.

2. TOWARDS AUTOMATED PRESERVATION STRATEGIES

Traditional data preservation deals with the reliable, distributed and replicated storage in order to prevent a phys-

Conference '10 Month 1-2, 2010, City, State, Country.

ical loss of the content bit-streams. Driven by the archival community, it has been recognized that data management plays a vital role for ensuring the long term accessibility and usability of digital information. A reference model for a preservation archive developed within the space data domain is provided by the Open Archival Information System (OAIS) [8]. In this paper, we present a system that specifically concentrates on the OAIS component "Preservation Planning".

One major difficulty for automated preservation arises from the diversity of digital data resources and methods to store, describe, and organize them. Large volumes of scientific data are generated by facilities and simulations and in fields like earth science, high energy physics, bioinformatics, or astronomy. Digital libraries and archives typically maintain holdings of complex digital objects (prints, databases, multimedia). These objects are organized by repository systems using (e.g. relational) data models which often implement rich metadata models that organize bibliographic, contextual, semantic, and technical information. However, these data models are individually designed and depend on the type of content a repository manages. In general, current repository systems usually do not provided methods for automated preservation actions and/or support preservation metadata. Preservation activities often depend on the individual data curators who are managing an archive, which makes digital preservation in many cases a manual and labor intensive task for archival institutions.

An important challenge that must be faced in this area is the automated preservation of large volumes of content, based on reproducible and verifiable preservation strategies. A typical problem is the preservation of data that requires software-based rendering in order to be human understandable (e.g. images, audio, video, scientific data). As a preservation strategy, one could for example try to preserve the rendering environment (emulation) or ensure the interpretability of the data (migration). However, a critical and nontrivial research problem is the automatic evaluation of the outcome of preservation actions [17].

Here, we describe a system that addresses two major goals: (1) The development of a set of defined preservation services for different preservation actions (e.g. migration, rendering, characterization) as well as strategies to describe, evaluate, and compare preservation results; (2) the development of an e-research environment that can access data from many different repositories systems/data stores, execute workflows on top of the data, record provenance and preservation metadata in a unified way, and deposit the experiment results. This paper concentrates on the latter aspects, work on the Planets preservation services has been reported in [15].

2.1 Example Use Case

Documents provide a good example for richness and diversity of digital objects. Besides rich formating information, numerous fonts, and digital signatures, documents can contain a variety of embedded objects (images, audio, video). Moreover, dozens of text processing applications and proprietary formats exist. For example, early packages like Word for Dos 5.5[1] can be utilized as a light-weight word processor on many mobile devices. A resulting document would not be interpretable with an up-to-date office software as it is. One would either needs to print the document into a postscript file using the original application or utilize an appropriate document conversion software. The document would not be lost but however require manual intervention in order to be meaningful on today's platforms. Here, we report a simple use-case for preserving a collection of outdated office documents that are endangered to become obsolete, using the IF workflow environment. We describe an example workflow and explain the various activities, involved services, and preservation components.

1. **Map and Register Data:** In order to make a digital collection available for experiments, it usually must be first retrieved from a managed repository environment. For this purpose, we map each obtained record to a generic data structure (called Planets Digital Object) that is stored within the digital object repository and registered with the data registry (shown in figure 1). The retrieval and mapping of digital repository records is facilitated by individual *Digital Object Managers*, described in section 3.3. Once the document collection is registered with the data registry, it can be browsed and referenced in the form of Planets URIs (e.g. *planets://dr/documents/mycollection/my.obj*).

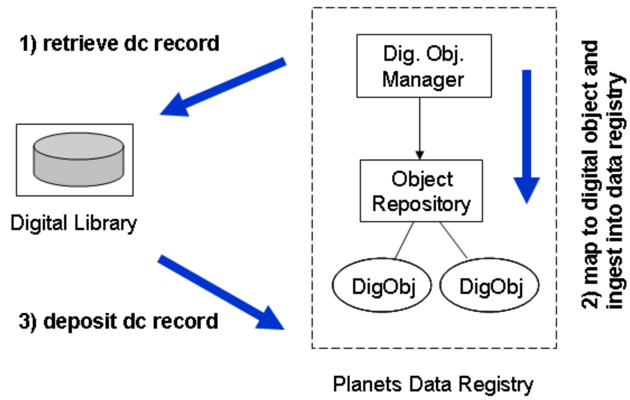

Figure 1: Registration of a dublin core (dc) record with the data registry. The metadata record is mapped to a Planets Digital Object and ingested into the object repository. Planets Digital Objects are made accessible through the Data Registry services.

2. **Develop a Preservation Strategy:** Developing a suitable preservation strategy (i.e. choosing migration pathways, tools, and parameters) can be a tedious and error-prone task. Planets provides a graphical decision support environment [2] that guides the user in generating an appropriate preservation plan. A resulting strategy could include the following steps: identify the exact format of each object within the document collection using the Droid[2] tool, migrate those that are in MS-Word formats to the ODF format using a conversion tool (provided by an ASP[3]), additionally migrate the ODF files to PDF/A, and characterize and compare the results using the XCL tool suite[4].

[1]http://download.microsoft.com/download/word97win/Wd55_be/97/WIN98/EN-US/Wd55_ben.exe

[2]http://droid.sourceforge.net/wiki/index.php/Introduction
[3]http://www.dialogika.de/
[4]http://planetarium.hki.uni-koeln.de/public/XCL/

3. **Choose and Configure a Workflow:** Once an executable workflow has been implemented for a specific scenario, it can be made available to other users by uploading it to the workflow repository in the form of a *workflow template*. Workflow templates specify an abstract execution process based on preservation components and decision logic. In order to create a concrete workflow instance, an XML-based configuration file that parameterizes the workflow template is submitted to the workflow execution engine (section 4). Such a workflow configuration file can be easily generated from the afore mentioned preservation strategy in order to instantiate the preservation workflow.

4. **Execution and Deposit:** During execution, the workflow engine orchestrates the involved preservation services based on the concrete workflow specification. Workflows are assembled from basic building blocks (preservation components) that are provided by the workflow API. Whenever an activity (e.g. a migration service) is being executed, the workflow component updates the digital object representation. For example, a migration event is added to the originating object, a new digital object instance is created, and a relationship between these objects is established. The result of a workflow execution is reflected by the newly generated and/or enriched digital objects within the data registry. The workflow results are accessible to the user for download via the data registry services and can be exported into XML documents. Recent work deals with approaches towards serializing the Planets digital objects instances using RDF graphs and the OAI-ORE model. This will allow users to deposit the experimentation results more easily in their institutional repositories.

2.2 Design Goals of the Framework

The Planets Interoperability Framework aims to provide a collaborative research environment for the evaluation of existing preservation tools as well as the development, sharing, and execution of novel preservation strategies. The system is based on a service-oriented architecture allowing the project participants to share applications, services, and workflow documents. A key aspect of the environment is the definition of the required preservation interfaces as well as a commonly shared digital object model. The Planets Testbed [10] currently supports a number of research deployments as well as a public and controlled environment for running benchmarks and experiments[5]. The primary focus of work on the IF however concentrates on implementing a service-oriented architecture for digital preservation, with the following goals:

- Provision of tools for preservation action like characterization, migration, and emulation as dynamically discoverable and executable services. Planets services conform to a set of conventions that provide technical and semantical interoperability allowing one to combine the individual services by constructing higher-level workflows.

- Development of common concepts (based on controlled vocabulary, ontologies, and registries) for preservation metadata that are produced and processed during an experiment.

- Establishment of an e-research environment based on a science portal, shared data, application services, and workflows documents. This includes support for dynamic resource management, workflow execution, and preservation metadata.

[5]http://testbed.planets-project.eu/testbed/

3. SYSTEM ARCHITECTURE

The Planets framework is designed as a research environment for digital preservation and does not aim to implement a preservation archive. The infrastructure provides a web portal framework that integrates a set of end user applications with a number of data repositories and a federation of grid/web and other services, such as services for data/metadata management, preservation, information, and workflow execution. We employ a generic data abstraction, called *digital object* in order to organize the different data sources through the Planets data registry. Provenance and other preservation information are automatically collected during workflow execution and expressed through the digital object model. The aim of the system is to provide an integrated environment that allows a community of researchers to collaboratively explore digital preservation strategies based on a number of shared resources.

3.1 Service Architecture

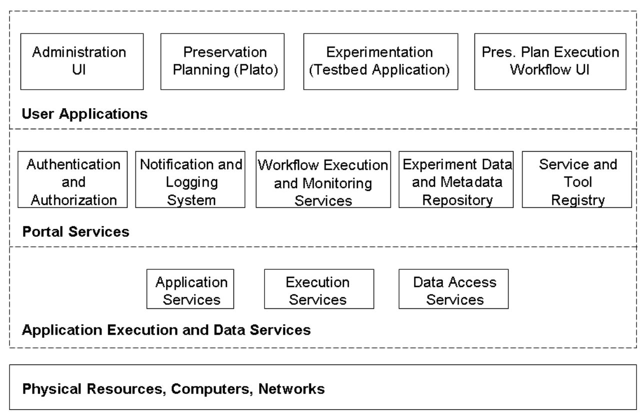

Figure 2: Service Architecture

The overall service architecture, as shown in figure 2, follows a classical tiered model consisting of an application layer, a set of portal services, and a number of execution and data services residing on the various physical resources. Each tier of the architecture communicates only with the neighboring tier based on Web service calls, notifications, or native invocations (if services reside in the same container). The application layer provides graphical user interfaces for utilizing the backend resources and administration usage, and is accessible though a single entry point provided by the web portal. The system facilitates single sign-on capabilities based on user name and password. The user credentials are propagated down to the lower system layers and mapped to pre-defined user accounts and roles. The *portal services* are consumed by the the application layer and implement a general API that reflects the capabilities of the gateway system. The current prototype implementation provides the following services: The *Authentication and Authorization* service maps user credentials to user accounts, hides the user's account details from other software components, and communicates with a token service. In its current version, the system solely relies on security mechanisms implemented by the Java EE[6] platform. The *Logging and Notification System* provides common logging capabilities for all framework components. The notification mechanism is implemented based on a simple publish-subscribe model and primarily used for monitoring process execution. Notifications are also used for logging, report generation or for sending email notifications. The *Workflow Service* provides a programmatic interface to the Planets workflow engine. The API allows one to choose from a set of preconfigured preservation workflow templates, configure an individual workflow instance (by choosing service endpoints and parameters), and schedule a workflow for execution upon a particular data set. The *monitoring* interface allows the retrieval of status information for a particular workflow instance. The *Data Registry* service allows clients to browse, register, and retrieve digital objects. Write access allowing a process to create and modify digital objects is restricted to local workflow components. The *Service Registry* provides fine-grained service discovery mechanisms including an extensible, schema- and ontology-based service categorization mechanisms. The registry maintains many different parameters including information about the service interfaces, the applications/tools, and supported parameters, as well as context information. The services that are provided at the *Application Service* layer must conform to defined, so called *level one*, interfaces that are supported by the workflow API. These interfaces define the way a preservation operation is being executed in terms of messaging, error handling, or parameterization. *Level one* services typically operate upon one or more underlying applications. The services can be implemented as local components, or operate upon remote commodity hardware; depending on the resource manager, they can provide access to high-throughput and clustered compute resources [16].

3.2 Registry Foundation

The Planets environment develops and integrates a number of technical registries that together define the vocabulary used by the preservation system. In the following we outline the foundation of information registries, the concepts they describe, as well as their interplay.

Metadata Definitions: The groundwork of the system is provided by a set of information registries that provide the required *Metadata Definitions* used by the digital object model. The definitions comprise of very general terms like formats and their properties as well as implementation-specific definitions like events and object properties. The concept of file formats is supported by the format registry. Different file formats are specified in the form of URIs (e.g. info:pronom/fmt/122 for EPS version 1.2) based on the PRONOM ID schema [4]. Information on the reading and rendering of digital data is provided by the OAIS *RepInfo* concept. A publicly available (not yet incorporated) registry is the Representation Information Registry provided by the UK

[6]java.sun.com/javaee/

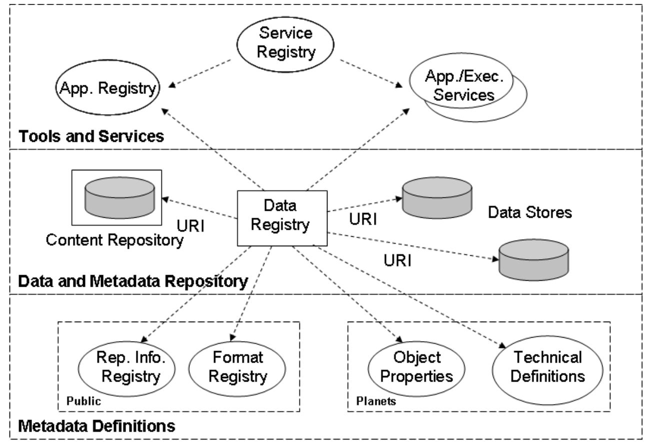

Figure 3: Digital objects are automatically enriched with metadata that is collected during a preservation workflow. This includes provenance information (e.g. on utilized services, tools, data) as well as preservation information like obtained format identifiers or content characteristics. The utilized vocabulary, as well as the services, tools and data are defined and unambiguously identified based on a set of registries.

Digital Curation Centre[7]. A local technical registry specifies the required concepts that are meaningful within the Planets system. Digital object properties provide a basic taxonomy and formal representation of digital object and format properties. An ontology that associates properties with different digital object types in order to facilitate the automated comparison and experiment evaluation has been built on top of these definitions. The *technical definitions* provide identifiers that are used by the digital object model, such as events, data types, and higher-level concepts.

Data Registry: The object repository which is a part of the *Data Registry* provides a space for storing digital object instances. Metadata that is expressed through the digital objects is initially generated during ingest and being modified by the workflow engine while performing preservation actions. Technical and preservation metadata that is contained within a digital object must be defined and be resolvable by the technical registries. Typical metadata concepts are checksum algorithms, filetype URIs, data types, associated properties, metrics, or events. The content bit-streams can be passed to the data registry based on a reference (a URI) or be directly encoded within the digital object.

Tools and Services: An important aspect of the provenance data that is being recorded during workflow execution are the tools and services that have been used to generate data or metadata. Beside the recording of timestamps, workflow and user identifiers, it is important to unambiguously identify the involved services and underlying tools. Planets maintains a service registry that generates an instance of a rich service descriptor for each preservation service endpoint. This information is generated once the service is registered with the preservation system and needs to be supported by a *service description* port type. Another important information for service selection and audit trail

[7]http://registry.dcc.ac.uk.

is the identification of the applications that a preservation service acts upon. It is therefore necessary to maintain a registry for all tool deployments including versions, environments, etc. The Planets service registry establishes the link between a preservation interface (e.g. for migrating one format into another), a concrete service deployment, and the underlying tool that implements the functionality.

3.3 Data Access and Integration

Access to repositories based on Planets Digital Objects is facilitated by individual Digital Object Manager implementations. These components are used to access existing repository systems from the Planets environment through a defined interface. During access, the individual data items are dynamically mapped to the Planets digital object model. For example, the title and description fields of a dublin core[8] record are directly mapped to the corresponding digital object attributes. Repositories may also embed technical information such as a checksum and algorithm within a record retrieved for example based on the OAI-PMH protocol. Metadata that is not interpreted by the digital object model can be still associated as tagged metadata chunks. This avoids any need to explicitly prescribe the nature of the high-level metadata representation. Content bit-streams are associated based on a reference (typically a URL) with a digital object or be directly embedded within the object, if required. Figure 4 shows a simplified class diagram of the digital object implementation.

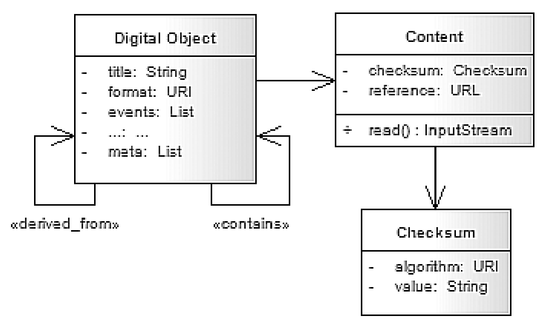

Figure 4: Simplified class diagram of the Planets Digital Object model showing a sample of properties, relationships, and referenced content.

The data registry, a component that consists of a *Digital Object Manager*, a *Digital Object Repository*, and a *Content Resolver* component is shown in figure 5. The digital object manager provides a hierarchical, browsable directory service that provides access to the object repository. It supports read and write access to the data registry, in contrast to other digital object managers that are used to retrieve records from remote repositories only. The interface is accessible to the Planets Workflow Execution Engine (WEE) and utilized to deposit/retrieve digital objects and associated experiment results. For example, a workflow may include the generation and ingest of new digital objects as well as facilitate the enrichment of existing objects. A workflow execution typically generates a result set that is returned to the client in order to trace and access the experiment results. Digital objects support a three-layer naming scheme comprising of a human readable name (title), a location independent name (permanent identifier), and a location dependent name (repository identifier). A digital object is referenced within the data registry using the Planets URI schema (e.g. *planets://dr/mynode/myobject*). The Planets IF supports an object repository that has been implemented based on the Apache Jackrabbit[9] framework, a reference implementation of the Content Repository for Java Technology API (JCR). Besides the digital object manager interface, this repository supports a *Content Resolver* service for directly accessing binary data as described in the next paragraph.

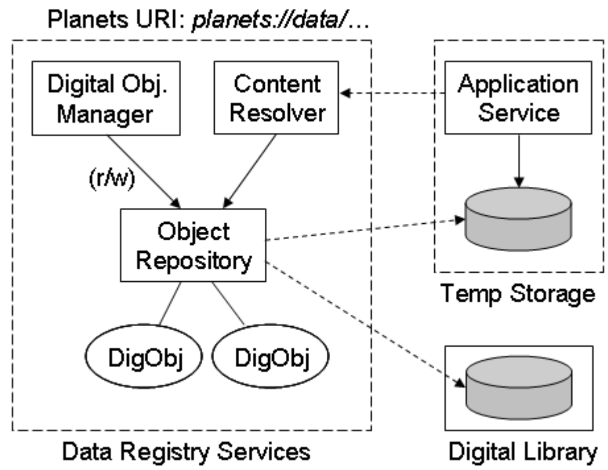

Figure 5: The Planets Data Registry provides a catalogue service allowing one to deposit, access, and organize Planets digital objects through its Digital Object Manager. It allows clients and services to interact with the registry based on exchanging metadata objects only. Binary content is typically passed based on references and automatically resolved and acquired during ingest.

An example data flow between the workflow engine, the data registry, and a preservation service is shown in figure 6. The transaction comprises of three activities (get object, migrate, and store) which are explained in the following: (1) The workflow engine retrieves a digital object (with a contained content reference) from the data registry using the data manager interface and passes it to a migration service; (2) after the preservation service has received the digital object, the content reference is resolved against the data registry (using the *ContentResolver*). The returned bit-stream is placed into the working directory of the preservation service. The service migrates the data and places the result file into a temporary storage that is accessible to the data registry; (3) finally, the service notifies the workflow engine and returns the result object (containing a reference to the generated content), which is added to the object repository by the workflow engine. During ingest, the content reference is automatically resolved and the content data is directly moved from the preservation service into the data registry. The *call-by-reference* mechanism avoids expensive copy operations between workflow engine and repository as well as unnecessary blocking and the loading of content into memory.

[8]http://dublincore.org/

[9]http://jackrabbit.apache.org/

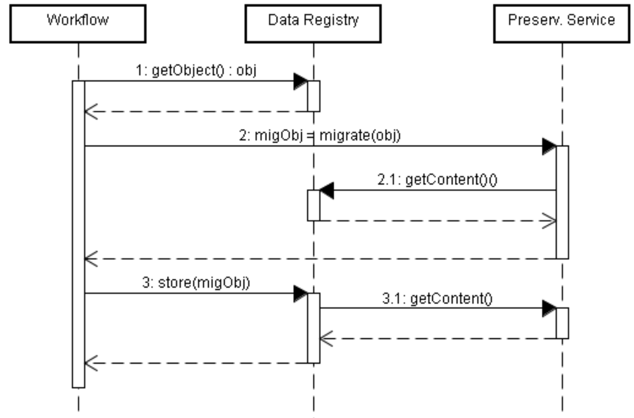

Figure 6: Example interaction between workflow engine, data registry, and a preservation service; the workflow engine orchestrates the exchange of metadata objects only; content is transfered on demand and based on repository references.

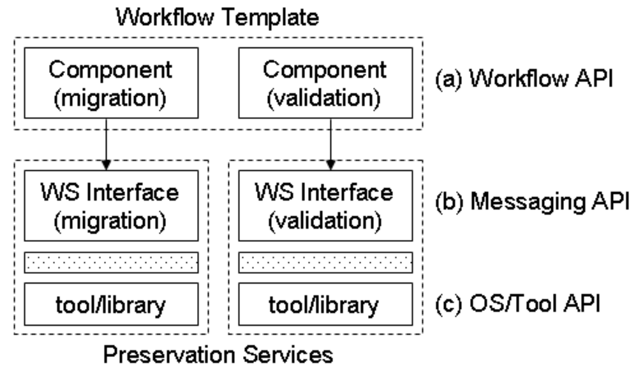

Figure 7: Interoperability Layers: IF workflow templates are composed from workflow components using a Java-based API (a). Preservation services expose defined Web services interfaces for preservation operations (b). Custom and/or generic wrappers implementation provide the glue code to the individual preservation tools (c).

4. WORKFLOW ENVIRONMENT

The Planets Interoperability Framework structures access to preservation tools into multiple abstraction and communication layers. Here, we briefly describe the IF workflow execution environment that provides a component for the flexible selection and execution of preservation tools.

4.1 Abstraction Levels

A preservation experiment comprises of a number of defined execution steps. Such preservation processes involve complex control logic as well as web service interactions and data model manipulations. A crucial requirement of the workflow environment is to allow data curators and archivists to assemble and deploy the preservation workflows they require, without forcing them to understand the underlying system. It is therefore important to abstract the complexity of the underlying architecture and its implementation details. This can be done by structuring the system into different abstraction layers and by employing a higher-level workflow language. Our approach provides a separation of concerns, so that not every party that intends to use/provide components of/to the preservation system needs to understand the entire communication and data model. In the context of Planets, we identified the following primary roles: (a) service/tool providers implement web service interfaces or simply describe/register an application that is provided by an generic execution service. At this level, metadata is not handled beyond basic status/error reports, (b) API developers integrate new interfaces by implementing higher-level workflow components. These components operate upon the lower-level services and encapsulate details like messaging and metadata. Workflow developers (c) create workflow templates that are assembled from a set of workflow components. Users/Experimenters (d) instantiate and execute workflows that are registered with the workflow environments, via a graphical interface.

4.2 Creating Workflows Templates

The Planets Interoperability Framework defines an extensible set of Web service interfaces for typical preservation actions like *migration*, *characterization*, *validation*, *comparing*, or *rendering*. The goal of this approach is to encapsulate the underlying preservation tools by a service interface in order to provide preservation operations in an unified and platform independent way. Figure 7 provides an overview of the different application layers and interfaces between them. Custom wrapper implementations provide the glue code between the Planets Web service interface and the underlying preservation tool or library. Planets workflows are built from high-level Java components and configured using XML descriptors. The workflow API allows workflow developers to easily create new workflow scenarios by assembling them from preservation components. These abstract workflows templates are made available using the workflow template repository, which is provided by the workflow environment.

4.3 Using the Workflow Environment

The IF workflow environment provides two basic services to client applications: a web service for browsing the workflow template repository as well as a service for the execution and monitoring of workflow instances. Using the workflow services, an experimenter can choose from a set of preconfigured preservation workflow templates, configure an individual workflow instance (by choosing service endpoints, parameters, variables), and schedule a workflow for execution upon a particular data set. The execution service provides basic monitoring functionalities to the user based on status inquiry and email-notification. The workflow engine is accessible to users via a generic web client as well as through web service and native interfaces to client applications. The results of a workflow are provided in the form of a report and can be traced and downloaded by accessing the data registry.

5. RELATED WORK

A clear demand for the integration of e-research infrastructures and repository technologies has been recognized in order to preserve scientific results and primary data [1].

The Genesi-DR[10] project for example provides discovery and access to a variety of dispersed geoscience data repositories. Data grid technology like SRB [12] can provide the underlying technology to create distributed preservation archives based on a virtual file system. An important aspect in this context is the storage of data in a reliable, distributed, and replicated way, as provided by the LOCKSS peer-to-peer network system [11]. The iRODS [13] environment extends SRB with an adaptive rule system and services to enforce data management policies. Other infrastructures concentrate on integrated digital library networks. The DARIAH [3] project develops a distributed data management infrastructure for connecting scholarly data archives and repositories with cultural heritage for the arts and humanities across Europe. The CLARIN project [18] aims to build a distributed infrastructure consisting of digital archives and repositories that provides access to language resources and tools through a common portal. D4Science[11] aims to provide a data-centric e-Infrastructures that is based on a digital libraries and grid technology.

6. CONCLUSIONS

We have presented a distributed e-research environment for the development of data preservation strategies that is being developed in the context of the EU project Planets. The project is driven by requirements for the long-term preservation of large volumes of digital materials faced by institutional libraries and archives. In this paper, we identify the main components and relationships of the infrastructure. We have outlined a service-oriented architecture that integrates preservation tools, data repositories, and information registries into a scalable research environment. The system is designed as a Science Gateway that operates upon distributed resources and provides a portal interface as a single point of access to end user applications. We employ a generic data model in order to organize the federated data sources and automatically collect provenance and other preservation information. Experimenters can develop and execute preservation strategies based on a set of defined preservation components and systematically validate the quality of the obtained results. The aim of this work is to provide an integrated environment that allows a community of researchers to collaboratively explore digital preservation strategies based on shared preservation services and data sources. The Planets system has been deployed across a number of European universities, research institutions, and private companies; at present it provides more then fifty tools as Planets services and facilitates access to distributed data repositories and digital collections provided by major European national archives and libraries.

Acknowledgments

Work presented in this paper is partially supported by European Community under the Information Society Technologies (IST) Programme of the 6th FP for RTD - Project IST-033789.

7. REFERENCES

[1] A. Aschenbrenner, T. Blanke, N. P. C. Hong, N. Ferguson, and M. Hedges. A Workshop Series for Grid/Repository Integration. *D-Lib Magazine*, 15(1/2), 2009.

[2] C. Becker, H. Kulovits, A. Rauber, and H. Hofman. Plato: a service oriented decision support system for preservation planning. In *Proc. of the JCDL '08*. ACM, 2008.

[3] T. Blanke and M. Hedges. Providing linked-up access to Cultural Heritage Data. In *Proc. of the ECDL 2008 Workshop on Information Access to Cultural Heritage*, 2008.

[4] A. Brown. The PRONOM PUID scheme: A scheme of persistent unique identifiers for representation information. *Digital Preservation Technical Paper 2*, 2005.

[5] A. Farquhar and H. Hockx-Yu. Planets: Integrated Services for Digital Preservation. *International Journal of Digital Curation*, 2(2), 2007.

[6] D. Gannon, B. Plale, M. Christie, L. Fang, Y. Huang, S. Jensen, G. K, S. Marru, S. L. Pallickara, S. Shirasuna, Y. Simmhan, E. Slominski, and Y. Sun. Service oriented architectures for science gateways on grid systems. In *In ICSOC*, pages 21–32, 2005.

[7] A. J. G. Hey and A. E. Trefethen. The data deluge: An e-science perspective. In *Grid Computing - Making the Global Infrastructure a Reality*, pages 809–824. Wiley and Sons, 2003.

[8] International Organization for Standardization . ISO Standard 14721:2003: Space Data and Information Transfer Systems Reference Model for an Open Archival Information System (OAIS). 2003.

[9] J. Alameda et al. The Open Grid Computing Environments collaboration: portlets and services for science gateways: Research Articles. *Concurr. Comput. : Pract. Exper.*, 19(6):921–942, 2007.

[10] A. Lindley, A. Jackson, and B. Aitken. A Collaborative Research Environment for Digital Preservation - the Planets Testbed. In *1st International Workshop on Collaboration tools for Preservation of Environment and Cultural Heritage (COPECH) at IEEE WETICE 2010*, Larissa, Greece, 2010.

[11] P. Maniatis, M. Roussopoulos, T. J. Giuli, D. S. H. Rosenthal, and M. Baker. The LOCKSS peer-to-peer digital preservation system. *ACM Trans. Comput. Syst.*, 23(1):2–50, 2005.

[12] A. Rajasekar, M. Wan, and R. Moore. MySRB & SRB: Components of a Data Grid. In *HPDC '02: Proceedings of the 11th IEEE International Symposium on High Performance Distributed Computing*, page 301, 2002.

[13] A. Rajasekar, M. Wan, R. Moore, and W. Schroeder. A prototype rule-based distributed data management system. *in: HPDC workshop on "Next Generation Distributed Data Management*.

[14] Rumsey, Abby S. (eds.). Sustainable Economics for a Digital Planet: Ensuring Long-Term Access to Digital Information. Technical report, Blue Ribbon Task Force on Sustainable Digital Preservation and Access, February 2010.

[15] R. Schmidt, R. King, A. Jackson, C. Wilson, F. Steeg, and P. Melms. A Framework for Distributed Preservation Workflows. In *Proceedings of The Sixth*

[10] www.genesi-dr.eu/

[11] http://www.d4science.eu/

International Conference on Preservation of Digital Objects (iPRES), San Francisco, USA, 2009.

[16] R. Schmidt, C. Sadilek, and R. King. A Service for Data-Intensive Computations on Virtual Clusters. In *Proceedings of The First International Conference on Intensive Applications and Services (INTENSIVE 2009)*. IEEE, 2009.

[17] M. Thaller, V. Heydegger, J. Schnasse, S. Beyl, and E. Chudobkaite. Significant Characteristics to Abstract Content: Long Term Preservation of Information. In *ECDL '08*, pages 41–49, 2008.

[18] T. Váradi, S. Krauwer, P. Wittenburg, M. Wynne, and K. Koskenniemi. CLARIN: Common Language Resources and Technology Infrastructure. In *Proc. of the Sixth International Language Resources and Evaluation (LREC'08)*, 2008.

[19] N. Wilkins-Diehr. Special Issue: Science Gateways—Common Community Interfaces to Grid Resources: Editorials. *Concurr. Comput. : Pract. Exper.*, 19(6):743–749, 2007.